
SSSSSSSSSSS YYY YYY SSSSSSSSSSS
SSSSSSSSSSS YYY YYY SSSSSSSSSSS
SSSSSSSSSSS YYY YYY SSSSSSSSSSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSS YYY YYY SSS
SSSSSSSSS YYY SSSSSSSSS
SSSSSSSSS YYY SSSSSSSSS
SSSSSSSSS YYY SSSSSSSSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSS YYY SSS
SSSSSSSSSSS YYY SSSSSSSSSSS
SSSSSSSSSSS YYY SSSSSSSSSSS
SSSSSSSSSSS YYY SSSSSSSSSSS

FILEID**SYSADJSTK

```
  SSSSSSSS  YY      YY   SSSSSSSS    AAAAAA    DDDDDDDD          JJ    SSSSSSSS  TTTTTTTTTT  KK      KK
  SSSSSSSS  YY      YY   SSSSSSSS    AAAAAA    DDDDDDDD          JJ    SSSSSSSS  TTTTTTTTTT  KK      KK
SS          YY      YY SS           AA    AA  DD      DD        JJ SS              TT      KK      KK
SS          YY      YY SS           AA    AA  DD      DD        JJ SS              TT      KK      KK
SS            YY  YY   SS           AA    AA  DD      DD        JJ SS              TT      KK    KK
SS            YY  YY   SS           AA    AA  DD      DD        JJ SS              TT      KK    KK
  SSSSSS        YY       SSSSSS     AA    AA  DD      DD        JJ   SSSSSS        TT      KKKKKK
  SSSSSS        YY       SSSSSS     AA    AA  DD      DD        JJ   SSSSSS        TT      KKKKKK
        SS      YY             SS   AAAAAAAAAA  DD      DD  JJ    JJ         SS    TT      KK    KK
        SS      YY             SS   AAAAAAAAAA  DD      DD  JJ    JJ         SS    TT      KK    KK
        SS      YY             SS   AA      AA  DD      DD  JJ    JJ         SS    TT      KK      KK    ....
        SS      YY             SS   AA      AA  DD      DD  JJ    JJ         SS    TT      KK      KK    ....
SSSSSSSS        YY     SSSSSSSS     AA      AA  DDDDDDDD      JJJJJJ   SSSSSSSS    TT      KK      KK    ....
SSSSSSSS        YY     SSSSSSSS     AA      AA  DDDDDDDD      JJJJJJ   SSSSSSSS    TT      KK      KK    ....

LL              IIIIII     SSSSSSSS
LL              IIIIII     SSSSSSSS
LL                II     SS
LL                II     SS
LL                II     SS
LL                II     SS
LL                II       SSSSSS
LL                II       SSSSSS
LL                II             SS
LL                II             SS
LL                II             SS
LL                II             SS
LLLLLLLLLL      IIIIII   SSSSSSSS
LLLLLLLLLL      IIIIII   SSSSSSSS
```

Table of contents

```
0000     1          .TITLE  SYSADJSTK - SYSTEM SERVICE ADJUST OUTER MODE STACK POINTER
0000     2          .IDENT  'V04-000'
0000     3
0000     4  ;
0000     5  ;****************************************************************************
0000     6  ;*                                                                          *
0000     7  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000     8  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9  ;*  ALL RIGHTS RESERVED.                                                    *
0000    10  ;*                                                                          *
0000    11  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16  ;*  TRANSFERRED.                                                            *
0000    17  ;*                                                                          *
0000    18  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000    20  ;*  CORPORATION.                                                            *
0000    21  ;*                                                                          *
0000    22  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000    24  ;*                                                                          *
0000    25  ;*                                                                          *
0000    26  ;****************************************************************************
0000    27  ;
0000    28  ; D. N. CUTLER 9-JAN-77
0000    29  ;
0000    30  ;
0000    31  ; MODIFIED BY:
0000    32  ;
0000    33  ;       V03-003 TMK0001         Todd M. Katz            19-Nov-1983
0000    34  ;               Change a BSBW (to EXE$EXPANDSTK) to a JSB.
0000    35  ;
0000    36  ;       V03-002 SRB0068         Steve Beckhardt         22-Feb-1983
0000    37  ;               Removed most of ACG0310.
0000    38  ;
0000    39  ;       V03-001 ACG0310         Andrew C. Goldstein,    31-Jan-1983  13:37
0000    40  ;               Fix stack adjustment when stack is expanded
0000    41  ;
0000    42  ;       02      RIH0031         RICHARD I. HUSTVEDT             6-AUG-1979
0000    43  ;               ADD CALL TO EXE$EXPANDSTK TO IMPLEMENT AUTOMATIC STACK
0000    44  ;               EXPANSION FOR USER MODE STACK.
0000    45  ;
0000    46  ; SYSTEM SERVICE ADJUST OUTER MODE STACK POINTER
0000    47  ;
0000    48  ; MACRO LIBRARY CALLS
0000    49  ;
0000    50
0000    51          $PSLDEF                         ;DEFINE PROCESSOR STATUS FIELDS
0000    52          $SSDEF                          ;DEFINE SYSTEM STATUS VALUES
0000    53
0000    54  ;
0000    55  ; LOCAL SYMBOLS
0000    56  ;
0000    57  ; ARGUMENT LIST OFFSET DEFINITIONS
```

```
                    0000     58 ;
                    0000     59
00000004            0000     60 ACMODE=4                     ;ACCESS MODE TO ADJUST STACK POINTER FOR
00000008            0000     61 ADJUST=8                     ;16-BIT SIGNED ADJUSTMENT VALUE
0000000C            0000     62 NEWADR=12                    ;ADDRESS OF LONGWORD TO STORE UPDATED VALUE
```

```
                                        .SBTTL  ADJUST OUTER MODE STACK POINTER
                                ;+
                                ; EXE$ADJSTK - ADJUST OUTER MODE STACK POINTER
                                ;
                                ; THIS SERVICE PROVIDES THE CAPABILITY TO ADJUST THE STACK POINTER FOR
                                ; A MODE THAT IS LESS PRIVILEGED THAN THE CALLING ACCESS MODE. IT CAN BE
                                ; USED TO LOAD AN INITIAL VALUE INTO THE SPECIFIED MODE'S STACK POINTER OR
                                ; TO ADJUST ITS CURRENT VALUE.
                                ;
                                ; INPUTS:
                                ;
                                ;               ACMODE(AP) = ACCESS MODE TO ADJUST STACK POINTER FOR.
                                ;               ADJUST(AP) = 16-BIT SIGNED ADJUSTMENT VALUE.
                                ;               NEWADR(AP) = ADDRESS OF LONGWORD TO STORE UPDATED VALUE.
                                ;                       IF THE INITIAL CONTENTS OF @NEWADR(AP) ARE NONZERO,
                                ;                       THEN THE VALUE IS TAKEN AS THE CURRENT TOP OF STACK.
                                ;                       ELSE THE CURRENT STACK POINTER FOR THE SPECIFIED MODE
                                ;                       IS USED.
                                ;
                                ; OUTPUTS:
                                ;
                                ;       R0 LOW BIT CLEAR INDICATES FAILURE TO ADJUST STACK POINTER.
                                ;
                                ;               R0 = SS$_ACCVIO - LONGWORD TO STORE UPDATED STACK POINTER
                                ;                       OR PART OF NEW STACK SEGMENT CANNOT BE WRITTEN BY
                                ;                       CALLING ACCESS MODE.
                                ;
                                ;               R0 = SS$_NOPRIV - SPECIFED ACCESS MODE IS EQUAL OR MORE
                                ;                       PRIVILEGED THAN CALLING ACCESS MODE.
                                ;
                                ;       R0 LOW BIT SET INDICATES SUCCESSFUL COMPLETION.
                                ;
                                ;               R0 = SS$_NORMAL - NORMAL COMPLETION.
                                ;-
                                
                                        .PSECT  Y$EXEPAGED
                                        .ENTRY  EXE$ADJSTK,^M<R2,R3,R4,R5,R6>
    55    0C AC    D0   0002          MOVL    NEWADR(AP),R5           ;GET ADDRESS TO STORE NEW STACK VALUE
53  04 AC  02    00   EF   0006      EXTZV   #0,#2,ACMODE(AP),R3     ;GET ACCESS MODE TO MODIFY STACK POINTER FOR
                 52   DC   000C      MOVPSL  R2                      ;READ CURRENT PSL
  53   52   02   16   ED   000E      CMPZV   #PSL$V_PRVMOD,#PSL$S_PRVMOD,R2,R3 ;PREVIOUS MODE MORE PRIVILEGED?
                 55   18   0013      BGEQ    60$                     ;IF GEQ NO
                      0015          IFNOWRT #4,(R5),40$             ;CAN NEW STACK VALUE BE WRITTEN?
            56   65   D0   001B      MOVL    (R5),R6                 ;GET SPECIFIED STACK VALUE
                 03   12   001E      BNEQ    10$                     ;IF NEQ VALUE SPECIFIED
            56   53   DB   0020      MFPR    R3,R6                   ;
          50    08 AC   32   0023  10$:  CVTWL   ADJUST(AP),R0         ;GET ADJUSTMENT VALUE
            56   50   C0   0027      ADDL    R0,R6                   ;CALCULATE NEW TOP OF STACK
            50   50   CE   002A      MNEGL   R0,R0                   ;ALLOCATION OF STACK SPACE?
                 17   15   002D      BLEQ    30$                     ;IF LEQ NO
            51   56   D0   002F      MOVL    R6,R1                   ;COPY NEW STACK VALUE
        52   FE00 8F   32   0032      CVTWL   #-^X200,R2              ;SET ADDITION CONSTANT
                      0037          IFNOWRT R0,(R1),40$,R3          ;CAN ALLOCATED STACK SEGMENT BE WRITTEN?
            51   52   C2   003D      SUBL    R2,R1                   ;UPDATE ADDRESS IN STACK
          50   6042   3E   0040      MOVAW   (R0)[R2],R0             ;UPDATE REMAINING LENGTH
                 F1   18   0044      BGEQ    20$                     ;IF GEQ MORE TO CHECK
            53   56   DA   0046  30$:  MTPR    R6,R3                   ;
```

```
                65   56   D0  0049   121          MOVL    R6,(R5)                 ;STORE NEW STACK VALUE
                50   01   3C  004C   122          MOVZWL  #SS$_NORMAL,R0          ;SET NORMAL COMPLETION
                          04  004F   123          RET                             ;
                53   03   D1  0050   124 40$:     CMPL    #PSL$C_USER,R3          ;IS THIS FOR USER MODE STACK?
                          11  0053   125          BNEQ    50$                     ;BR IF NOT
                     3E   BB  0055   126          PUSHR   #^M<R1,R2,R3,R4,R5>     ;SAVE REGISTERS
                52   51   D0  0057   127          MOVL    R1,R2                   ;STACK BASE ADDRESS
00000000'       EF        16  005A   128          JSB     EXE$EXPANDSTK           ;AUGMENT STACK TO MAKE ACCESSIBLE
                     3E   BA  0060   129          POPR    #^M<R1,R2,R3,R4,R5>     ;RESTORE REGISTERS
                   B6 50  E8  0062   130          BLBS    R0,5$                   ;REPEAT CHECKS
                          04  0065   131          RET                             ;RETURN ERROR CODE
                50   0C   3C  0066   132 50$:     MOVZWL  #SS$_ACCVIO,R0          ;SET ACCESS VIOLATION
                          04  0069   133          RET                             ;
                50   24   3C  006A   134 60$:     MOVZWL  #SS$_NOPRIV,R0          ;SET NO PRIVILEGE
                          04  006D   135          RET                             ;
                              006E   136
                              006E   137          .END
```

```
ACMODE                          = 00000004
ADJUST                          = 00000008
EXE$ADJSTK                        00000000 RG    02
EXE$EXPANDSTK                     ******** X     02
NEWADR                          = 0000000C
PSL$C_USER                      = 00000003
PSL$S_PRVMOD                    = 00000002
PSL$V_PRVMOD                    = 00000016
SS$_ACCVIO                      = 0000000C
SS$_NOPRIV                      = 00000024
SS$_NORMAL                      = 00000001
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
Y$EXEPAGED	0000006E (110.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.10	00:00:00.43
Command processing	112	00:00:00.60	00:00:04.80
Pass 1	202	00:00:04.25	00:00:14.53
Symbol table sort	0	00:00:00.69	00:00:02.12
Pass 2	42	00:00:00.76	00:00:05.12
Symbol table output	3	00:00:00.02	00:00:00.03
Psect synopsis output	2	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	392	00:00:06.45	00:00:27.07

The working set limit was 1200 pages.
22668 bytes (45 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 451 non-local and 7 local symbols.
137 source lines were read in Pass 1, producing 16 object records in Pass 2.
10 pages of virtual memory were used to define 9 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYS.OBJ]LIB.MLB;1	1
_255DUA28:[SYSLIB]STARLET.MLB;2	5
TOTALS (all libraries)	6

516 GETS were required to define 6 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:SYSADJSTK/OBJ=OBJ$:SYSADJSTK MSRC$:SYSADJSTK/UPDATE=(ENH$:SYSADJSTK)+EXECML$/LIB
```

0381 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

SSDEF
LIS

SYSACPFDT
LIS

SYSADJSTK
LIS

SYSASCEFC
LIS

SYSBRKTHR
LIS

SYSASSIGN
LIS

SWAPPER
LIS

SYSASTCON
LIS

SVAPTE
LIS

SYSADJWSL
LIS